Computer Architecture
ISA

Examples on Addressing Modes

Basic Instruction

- **Add:** add two numbers

- **Mul:** multiply two numbers
- **Move**: move data between two places
- **Load:** load data from memory to register

- **Store:** store data from registers to memory

- **Push and Pop** for stack operations

Register Types: Assume 16-bit registers

- Accumulator register (ACC)
- Instruction register (IR)
- Program Counter (PC)
- Address register (AR)
- General purpose registers (R0 t0 R6)

Write an assembly language program

- To write a program using ISA we should know the addressing mode of the processor.

- The operands fetch operation will be different based on the addressing mode.

- **Steps for write an assembly program:**
  1. Specify the addressing mode type.
  2. Convert the statement to assembly language.

Example

- Write an assembly code for execute the following statements **using accumulator addressing mode, register addressing mode, indirect mode and register indirect** and store the result in memory which has the address stored in AR:

  - F=A+B+C
  - X=A+(B*4)
  - Z=A*B*C+7

F=A+B+C

Using Accumulator addressing mode

- Load A
- Add B
- Add C
- Store M[AR]

F=A+B+C

Using Register addressing mode

- Load R0, A
- Load R1, B
- Load R2, C
- Add R0,R1
- Add R0,R2
- Store M[AR],R0

F=A+B+C

Using indirect addressing mode

- Load R0, (A)
- Load R1, (B)
- Load R2, (C)
- Add R0,R1
- Add R0,R2
- Store M[AR],R0

F=A+B+C

Using register indirect addressing mode

- Load R0, (R3)
- Load R1, (R4)
- Load R2, (R5)
- Add R0,R1
- Add R0,R2
- Store M[AR],R0

Indicate the used addressing mode for each of the following instructions:

- Move R0,R1….
- Add 5….
- Move R0, (A)……
- Add (R0), (R1)…….
- Add A…….

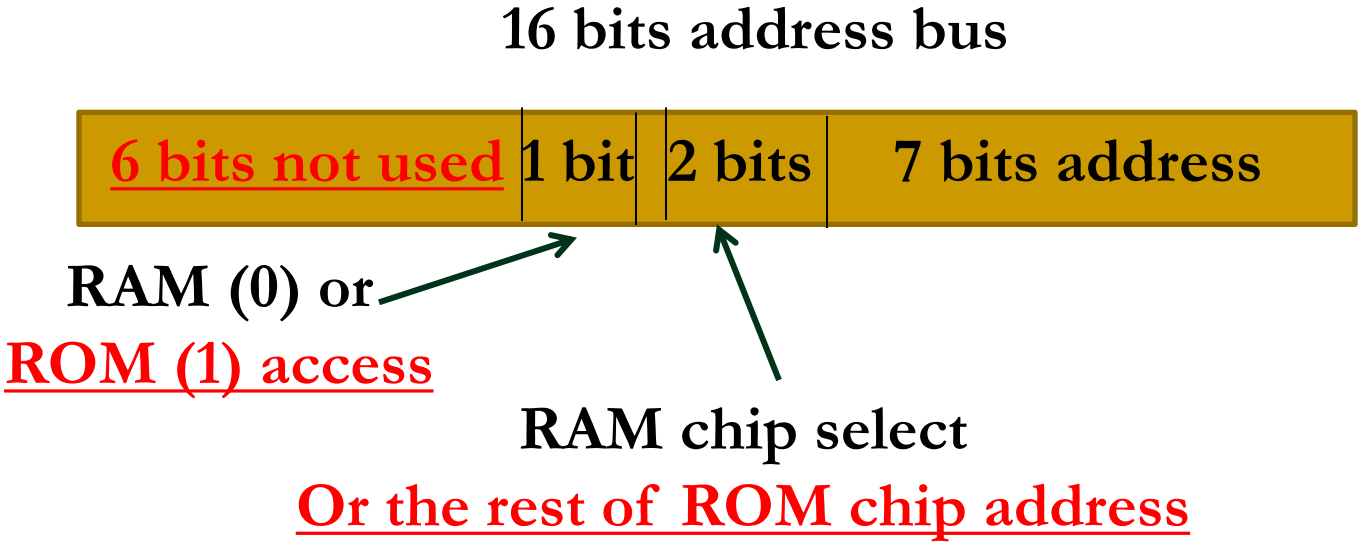
- Move R0,R1….Register mode
- Add 5….Immediate mode
- Move R0, (A)……Indirect mode
- Add (R0), (R1)…….Register Indirect mode
- Add A…….Accumulator and Direct mode

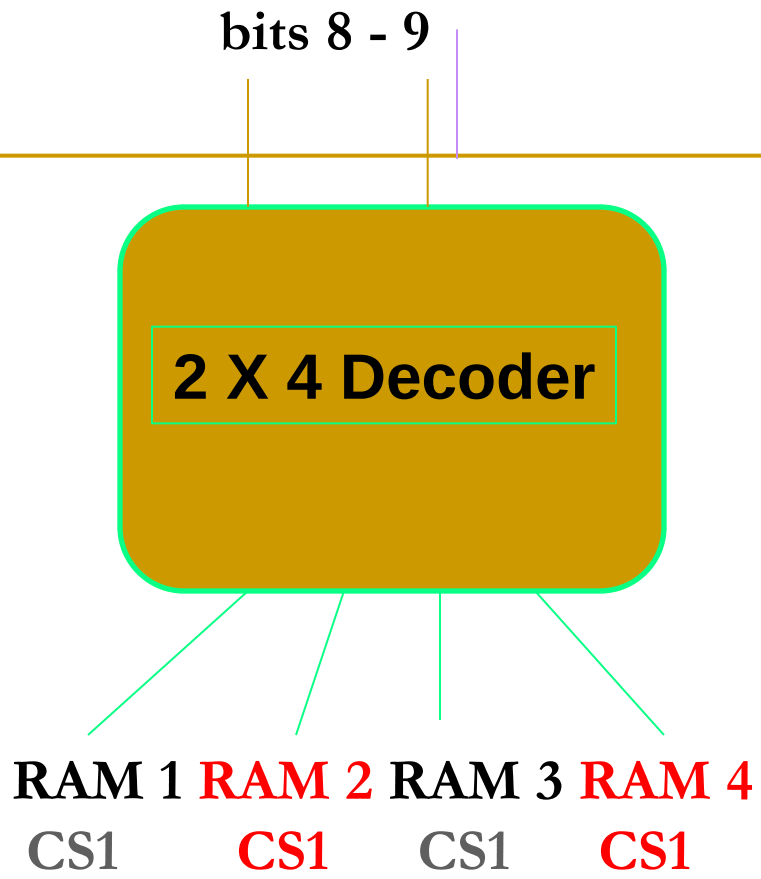
Practical Example : How CPU deal with RAM and ROM

Assume that a computer system has 512 bytes of RAM and 512 bytes of ROM. How CPU deal with RAM and ROM? Draw memory organization block diagram.

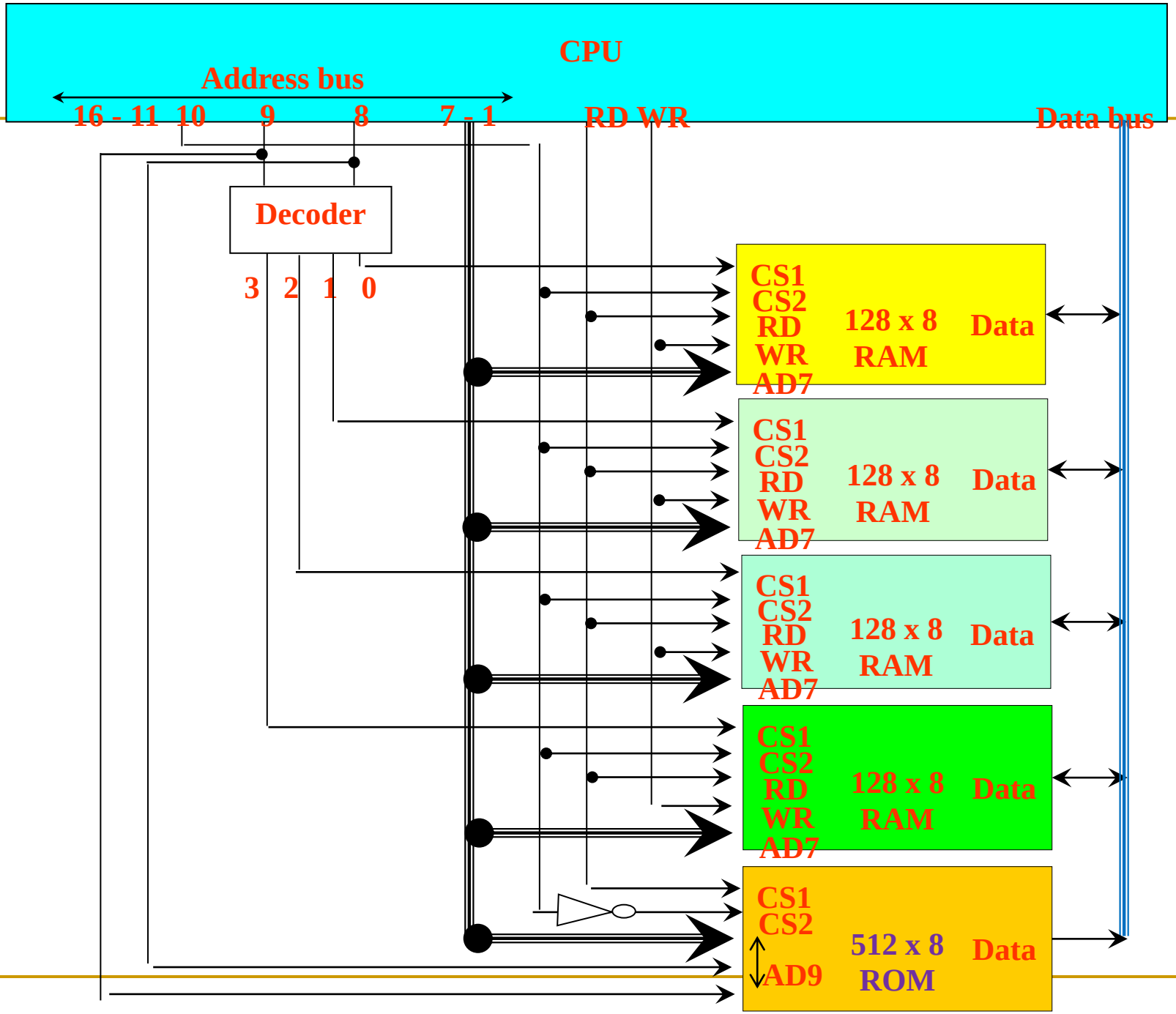
Assume that a computer system has:

- 512 bytes of RAM (we use 4 blocks of RAM of the same type 128X8)
- 512 bytes of ROM (single block).





We use a decoder to select one of the 4 RAM chips using bits 8 – 9 by connecting the output to CS1 of the RAM Chip


CPU
Address bus
16 - 11
10
9
8
7 - 1
RD WR
Data bus
Decoder
3
2
1
0
CS1
CS2
RD
WR
AD7
128 x 8
RAM
Data
CS1
CS2
RD
WR
AD7
128 x 8
RAM
Data
CS1
CS2
RD
WR
AD7
128 x 8
RAM
Data
CS1
CS2
RD
WR
AD7
128 x 8
RAM
Data
CS1
CS2
AD9
512 x 8
ROM
Data

Component	Hexadecimal address	Address bus									
		10	9	8	7	6	5	4	3	2	1
RAM 1	0000-007F	0	0	0	X	X	X	X	X	X	X
RAM 2	0080-00FF	0	0	1	X	X	X	X	X	X	X
RAM 3	0100-017F	0	1	0	X	X	X	X	X	X	X
RAM 4	0180-01FF	0	1	1	X	X	X	X	X	X	X
ROM	0200-03FF	1	X	X	X	X	X	X	X	X	X

$(0000\ 0000\ 0000\ 0000)_2 = (0000)_{16}$
$(0000\ 0000\ 0111\ 1111)_2 = (007F)_{16}$

$(0000\ 0000\ 1000\ 0000)_2 = (0080)_{16}$
$(0000\ 0000\ 1111\ 1111)_2 = (00FF)_{16}$

$(0000\ 0001\ 0000\ 0000)_2 = (0100)_{16}$
$(0000\ 0001\ 0111\ 1111)_2 = (017F)_{16}$

Thanks for your attention